Murdoch UNIVERSITY

MURDOCH RESEARCH REPOSITORY

http://dx.doi.org/10.1109/ICASSP.1990.116049

Fung, P.W., Grebbin, G. and Attikiouzel, Y. (1990) Contextual classification and segmentation of textured images. In: International Conference on Acoustics, Speech, and Signal Processing, ICASSP-90, 23 - 26 April, Albuquerque, NM. USA, pp. 2329 - 2332.

http://researchrepository.murdoch.edu.au/19605/

Copyright © 1990 IEEE

Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

CONTEXTUAL CLASSIFICATION AND SEGMENTATION OF TEXTURED IMAGES

P. W. Fung, G. Grebbin, Y. Attikiouzel

Department of Electrical & Electronic Engineering
The University of Western Australia, Nedlands, Western Australia 6009

ABSTRACT

A new algorithm which combines the merits of statistical classification and estimation theory based approaches is proposed for textured image segmentation. The texture regions are modelled by noncausal Gaussian Markov Random Fields(GMRF). The algorithm comprises of two stages: statistical classification and edge estimation. In the first stage, the image is partitioned into small blocks of pixels. GMRF parameter estimates are extracted as the feature vector for each block. A maximum likelihood spatial classifier, which explores the class conditional correlation properties among neighbouring feature vectors, is proposed for classifying each block into one of m possible texture classes. The result of classification is a coarse segmented image. The locations of the edges are estimated in the second stage using a line by line maximum likelihood edge estimation technique. Each detected edge sequence is further modelled as an autoregressive(AR) process and processed by a Kalman filter in order to smooth the detected boundary.

1. INTRODUCTION

The segmentation of an image into regions that are homogeneous with respect to texture property is an important task in many image processing applications. This problem has received considerable attention during the last decade. Most segmentation algorithms are based on statistically classifying pixels, or blocks of pixels, into one of the texture classes after the transforming the intensity space into feature space. Since texture is basically a local property, these methods represent a natural approach toward textured image segmentation. Many different types of features has been used in texture segmentation. These include the co-occurrence matrix[1,2], runlength statistics[3], power spectrum[4], and parameters from statistical image models[5,6]. Generally, texture features are multidimensional, thus complex classification methods, such as Bayes' method and the nearest neighbour method, are used to classify pixels in feature space. Furthermore, the existence of a priori texture information is usually assumed in these methods. However, most of these methods suffer from the following drawbacks :

(1) The segmented regions are not guaranteed to be connected. Isolated spots or spurious regions often exist.
(2) To reduce computational burden, classification is often applied to sets of small blocks of pixels that divide the image. This results in poor localization of detected boundaries with respect to actual boundaries.

Recently, attention has focussed on estimation theory based segmentation techniques[7-9]. These techniques aim at extracting the correlation structures of the texture field in the image and representing each texture field by a spatial interaction model. In such cases, the task of segmentation is formulated as either a maximum likelihood estimation(ML) problem or a maximum a posteriori(MAP) problem. Dynamic programming,or deterministic or stochastic relaxation techniques are often applied to achieve optimal solutions to such problems. These techniques usually provide very accurate segmentation, but their computation cost is enormous, especially when the size of the image and the number of possible texture types are large.

This paper presents a new algorithm for textured image segmentation which retains the merits of the statistical classification and estimation theory based approaches while avoiding most of their problems. The different textures which may exist in the image are assumed to be samples of homogeneous random fields that are characterized by 2-D noncausal Gaussian Markov Random Fields(GMRF's). The algorithm is supervised in that a priori information about the number of possible texture classes and the texture features of each class is given. The algorithm consists of two stages: statistical classification and edge estimation. In the first stage, the image is divided into small disjoint D×D blocks of pixels, with each block being the central core of an L×L window. Texture features are extracted from each window and are used to represent the characteristics of its central core. Parameter estimates of the GMRF in the window are taken as texture features in this case. The segmentation problem is then considered as a classification problem which labels each block as one of the texture classes according to the characteristic of its features. Since adjacent windows overlap, texture features of neighbouring blocks are conditionally correlated. A maximum likelihood spatial classifier which utilizes the spatial information among neighbouring blocks is developed. This approach has been shown to provide significant improvement in texture classification over any non-spatial classifier. The classification process generates a coarse segmentation, from which boundary regions can be identified. The exact location of the edges in these boundary regions are estimated, in the second stage, using an edge estimation approach. Edges are estimated on a line by line basis using a maximum likelihood estimation technique. To account for the second dimension of the image, a Kalman filter is used to provide smooth edge contours. The proposed 1-D edge estimation process provides much higher efficiency in computation than its 2-D counterparts.

In the next section, a brief introduction on the structures of the GMRF texture model is presented. Section 3 considers the development of the ML spatial classifier. Section 4 gives the design and computational procedures for edge detection and smoothing. Finally, experimental results on real textured images are presented in section 5.

2. GAUSSIAN MARKOV RANDOM FIELD (GMRF)

The use of the GMRF to model image textures has been studied extensively for a number of years[6,9]. The properties of GMRF can be summarized as following: let $Y = \{y_r\}$ be an L×L texture random field defined over a square window $\Omega = \{ r = (i,j);\ 1 \leq i,j, \leq L\}$, with y_r the field at pixel r; the field Y is said to be a GMRF if its conditional likelihood function can be written as

$$p(y_r \mid Y_{(r)}) = [2\pi\sigma^2]^{-1/2}\exp\{-(1/2\sigma^2)[y_r-\mu-\sum_{v\in N}\beta_v(y_{r-v}-\mu)]^2\} \quad (1)$$

where β_v indicates the spatial interaction between pixel r and r-v, N is a neighbour set, $Y_{(r)} = \{y_v;\ v \in \Omega,\ v \neq r\}$, μ is the sample mean, and σ^2 is the conditional variance. If Y is noncausal, N is symmetric. It can also be shown that the GMRF can be

CH2847-2/90/0000-2329 $1.00 © 1990 IEEE

equivalently represented by a linear interpolative equation:

$$y_r = \sum_{v \in N} \beta_v (y_{r-v} - \mu) + \mu + e_r \qquad (2)$$

where the zero mean stationary Gaussian noise sequence $\{e_r\}$ has the following properties:

$$\begin{aligned} E[e_r e_{r-v}] &= -\beta v \sigma^2 && v \in N; \\ &= \sigma^2 && v = 0; \\ &= 0 && \text{otherwise} \end{aligned}$$

The GMRF is completely parameterized by $\Theta = \{ \{\beta_r, r \in N\}, \mu, \sigma^2 \}$, and various methods have been proposed to obtain such estimates[6,10]. In our study, the least square(LS) estimates of Θ are selected as texture features. Details of the estimation algorithm can be found in [6]. Maximum likelihood(ML) estimation of these parameters is also possible. However, LS estimation provides a simpler and faster solution to the problem.

3. MAXIMUM LIKELIHOOD SPATIAL CLASSIFIER

In the classification phase of the algorithm, the image is first scanned, from left to right and top to bottom, by an LxL window. The window slides over the image with its centre positioned on small blocks of DxD pixels that partition the entire image. During scanning, the GMRF parameters Θ are extracted from the region enclosed by the window. These parameters become the features of the central block of that window. A feature image FI is then created, with each pixel in the FI image representing a DxD block of pixels in the original image. The feature Θ for each block is assigned as the pixel "value" of the FI image. Initial segmentation could then be preformed by classifying each pixel in the FI image into one of m possible classes. After classification, the segmented FI image is mapped onto the original image. The result is a coarse segmentation of the original image. There are several existing techniques for statistical classification of features in multidimensional space. For example, denoting the feature set Θ at location $s \in \Omega$ as vector $X(s) = [x_1(s), x_2(s), \ldots, x_q(s)]^T$, and $P(X(s) | \omega_k)$ the class conditional density function for class ω_k, $k = 1,\ldots,m$, the decision rule for the ML classifier is given by:

$$\text{if } P(X(s) | \omega_\ell) = \max_{k=1,\ldots,m} P(X(s) | \omega_k) \qquad (3)$$

then classify $X(s)$ into class ω_ℓ. $\Omega = \{ s \in (i,j), 1 \le i,j, \le M\}$ is the coordnate set of the FI image. If the $X(s)$'s are assumed to be spatially uncorrelated Gaussian random vectors, the decision rule can be simplified as follows

$$\begin{aligned} &(X(s) - \overline{X}_\ell)^T \Sigma_\ell^{-1} (X(s) - \overline{X}_\ell) + \log |\Sigma_\ell| \le \qquad (4) \\ &(X(s) - \overline{X}_k))^T \Sigma_k^{-1} (X(s) - \overline{X}_k) + \log |\Sigma_k| \quad k = 1, \ldots, m \end{aligned}$$

where $\overline{X}_k(s)$ and Σ_k are, respectively, the mean vector and covariance matrix of class ω_k, and can be estimated from a training set of each class. Further simplification can be achieved if Σ_k is a diagonal matrix, that is, the elements in $X(s)$ are uncorrelated. In such cases, the decision rule becomes

$$\begin{aligned} &\sum_{j=1}^{q} \{\log\sigma^2_{j,\ell} + (x_j(s) - \bar{x}_{j,\ell}(s))^2 / \sigma^2_{j,\ell}\} \le \\ &\sum_{j=1}^{q} \{\log\sigma^2_{j,k} + (x_j(s) - \bar{x}_{j,k}(s))^2 / \sigma^2_{j,k}\} \qquad (5) \end{aligned}$$

where $\bar{x}_{j,k}$, $\sigma^2_{j,k}$ are, respectively, the mean value and variance of the j element in $X(s)$ of class ω_k.

The major drawback of the above classification scheme is that no contextual, or spatial information, is utilized in class identification. Since the feature estimation windows for adjacent blocks overlap, the extracted features are spatially correlated. Thus, two FI image pixels in spatial proximity are class conditionally correlated, and spatial information could be incorporated into the classification process. By so doing, we assume that spatial variation in the FI image can be modelled by

$$X(s) = \overline{X}_k + \sum_{r \in N} \Phi_k(r)(X(s-r) - \overline{X}_k) + W_k(s) \qquad s \in \Omega_k \quad (6)$$

where N is a neighbour set, Ω_k is a subset of Ω in which $X(s)$, $s \in \Omega_k$, belongs to class ω_k, and $\Phi_k(r)$, $r \in N$, is a qxq matrix which describes the spatial correlation of class ω_k. Also, $W_k(s)$, $s \in \Omega_k$, is a Gaussian white noise field which satisfies

$$E[W_k(s)] = 0$$

$$\begin{aligned} E[W_k(s) W^T_k(r)] &= R_k && \text{if } s = r \\ &= 0 && \text{if } s \ne r \end{aligned}$$

Other properties of $X(s)$ are

$$E[X(s) | \{X(s), s \in \Omega_k\}, \omega_k] = \overline{X}_k + \sum_{r \in N} \Phi_k(r)(X(s-r) - \overline{X}_k)$$

$$COV[(X(s) - \overline{X}_k) | \{X(s), s \in \Omega_k\}, \omega_k] = R_k$$

$\Phi_k(r)$, $r \in N$, and R_k can be estimated directly from training samples of class k. The estimation can be simplified dramatically if $\Phi_k(r)$ is assumed to be diagonal. In such cases, the spatial correlation of each element in $X(s)$ can be estimated independently of the others. A Least squares estimation method can be used to obtain such estimates. The method is straight forward, and is described in [11].

With the proposed model, the decision rule of (4) can be modified into

$$Y_\ell^T(s) R_\ell^{-1} Y_\ell(s) + |R_\ell| \le Y_k^T(s) R_k^{-1} Y_k(s) + |R_k| \quad (7)$$

where $Y_k(s) = X(s) - \overline{X}_k - \sum_{r \in N} \Phi_k(r)(X(s-r) - \overline{X}_k)$

Note that eqn. (7) clearly shows the dependency of the decision rule on the spatial correlation $\Phi_k(s)$. The incorporation of this dependency into the decision rule provides significant improvement in classification, as will be seen in the experiments.

4. EDGE ESTIMATION

The second stage of the algorithm estimates edges in regions where the exact texture boundaries are believed to be located. Such boundary regions can be identified very easily after the coarse segmentation is obtained. One simple way of finding boundary regions from the coarse segmented image is to partition the image into small disjoint square windows. The texture classes within each window are examined. If more than two texture classes exists, the window is classified as mix texture type and edge estimation will then be applied to locate the exact border in that window. The advantage of this approach is threefold. First, since the edge estimations are performed only in the boundary windows, some computation is saved. Second, when the window is small, most of the boundary windows will be composed of only two texture types, hence, computational complexity is greatly reduced. Third, since the texture types in each window are known *a priori*, edge estimation is more efficient and accurate. In the following sections, a line by line edge estimation technique is presented. At first, the number of texture types in each boundary window is limited to two. Then, extensions to more than two texture types will be discussed.

4.1 Maximum Likelihood Edge Estimator

Let an NxN boundary window consist of two texture classes, ω_1 and ω_2. Figure 1 shows some possible examples of such a window. Note that the boundary across the window is assumed to be single-valued with respected to either the horizontal or vertical axis. This is often the case when the window size is small and the texture boundary is sufficiently smooth in nature.

Thus, it is important to choose between horizontal and vertical scanning directions for each window in the edge estimation process. One method of making this determination is to run the edge estimator in both horizontal and vertical directions. Linear mean value functions are fitted to the estimated boundaries. The most satisfactory direction is that which makes the smallest angle with the function. The following discussion will assume that the horizontal scanning direction has been chosen. Similar arguments can be applied to the vertical direction.

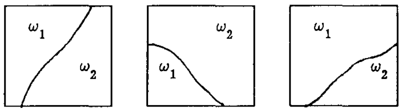


Fig. 1

The window is first scanned in a line by line fashion. The edge pixel which separates two texture fields along a scan line is estimated using a maximum likelihood estimator. Assuming $\{y_s, s = (i,j), 1 \le j \le N\}$ to be a GMRF alone the ith line which consists of two texture classes, ω_1 and ω_2, an edge pixel is estimated at position $r = (i,n)$ if it satisfies

$$P(\{y_s\} \mid r = (i,n),\omega_1,\omega_2) = \max_{j=1,\dots,N} P(\{y_s\} \mid k = (i,j),\omega_1,\omega_2) \quad (8)$$

where $P(\{y_s\} \mid k = (i,j),\omega_1,\omega_2)$ is the joint likelihood function of $\{y_s\}$ in the ith line when given ω_1 and ω_2. Maximization of (8) is not an easy task since the joint likelihood function is quite complicated. However, if we assume that the noise samples along the line are statistically independent, then the joint likelihood function simplifies to

$$P(\{y_s\} \mid k=(i,j),\omega_1,\omega_2) = \prod_{v=(i,1)}^{v=(i,j)} P(y_v \mid \omega_1) \prod_{u=(i,j+1)}^{u=(i,L)} P(y_u \mid \omega_2) \quad (9)$$

with $P(y_r \mid \omega_i)$, $i = 1$ or 2, given by eqn.(1).

Although this is not a true likelihood function, this pseudo likelihood function is very effective for edge estimation and its computation is extremely efficient. Furthermore, there is the possibility that a scan line may consist of a single texture, belonging to either ω_1 or ω_2. In such cases, a hypothesis test must be established to determine whether the texture field $\{y_s\}$ along a line is a single or mixed texture type. The hypotheses to be tested are

H0 : the line holds a single texture from class ω_i, $i = 1$ or 2
H1 : the line composes of texture ω_1 and ω_2

To this end, a generalized likelihood ratio test(GLR) is performed. The GRL test computes the likelihood ratio

$$\Lambda = P(\{y_s\} \mid H1,\ r = \hat{r}) / P(\{ys\} \mid H0)$$

where $\hat{r}$ is the ML estimate of the edge position under the hypothesis H1. Hence, the decision becomes

if $\log \Lambda > \epsilon$ then accept H1 else accept H0

Thus, if H1 is accepted, $\hat{r}$ is taken as the edge position alone that line.

When the boundary window consists of more than two texture types, a hierarchical edge estimation scheme can be employed. This scheme consists of successive applications of the ML edge estimator on the window. The image is first considered as composed of two texture regions and the ML edge estimator is used to segment the image into two regions; then each region is segmented into two region types and so on. Thus, using the ML edge estimator alone, boundary windows composed of m texture types can be estimated by m-1 applications of the edge estimator. However, with a small window size, m is unlikely to be greater than four.

4.2 Boundary Smoothing Using Kalman Filter

The line by line edge estimation process yields a sequence of position coordinates $\{z_i, i = 1,.. N\}$ of which z_i is the position of the estimated edge in the ith line. To take account of the second dimension of the window and to correct for any detection error, the sequence $\{z_i\}$ is smoothed by a Kalman filter. To establish the computation procedure, we need to develop a statistical model of the edge sequence. Assuming $\{z_i\}$ is governed by a first order AutoRegressive (AR) model process with a linear mean function, the random variable z_i is therefore given by

$$z_i = m(i) + r_i + \alpha_i$$

where r_i is a zero mean Gaussian random variable satisfying the first order AR process

$$r_{i+1} = \phi r_i + \delta_i$$

Also, α_i and δ_i are white noise sequences with variance ρ^2 and ν^2 respectively. $m(i) = a_0 + a_1 i$ represents the linear mean function of the boundary, and can be estimated by a least square fit. Parameters ρ^2, ν^2 and ϕ can be preassigned or estimated directly from $\{z_i\}$. Once all parameters are determined,a Kalman filter can be employed for the smoothing process. The equations for computing the smooth version of $\{z_i\}$ are[12]

$$\begin{array}{lll}
P_0 = 0 & r_0 = 0 & \\
P_{j+1/j} = \phi^2 P_j + \nu^2 & & j = 0,\dots,N \\
P_j = \rho^2 P_{j/j-1}/(P_{j/j-1} + \rho^2) & & j = 1,\dots,N \\
K_j = P_j/\rho^2; & & j = 1,\dots,N \\
r_j = \phi r_{j-1} + K_j[z_j - \phi r_{j-1}] & & j = 1,\dots,N
\end{array}$$

After filtering, the linear mean m(i) must be added to r_i to obtain the final position coordinates of the edge pixels. Apart from smoothing the edge sequence within each boundary window, the filtering process also eliminates any boundary discontinuity between adjacent windows.

5. EXPERIMENTAL RESULTS

The segmentation algorithm has been tested on a series of images containing different regions of natural textures. Figures 2-4 illustrate some experimental examples. Each image has 256×256 pixels with 256 gray levels. A noncausal GMRF of order 3 is used to model the texture fields. Each image is divided into blocks of 8×8 pixels during the classification phase of the segmentation algorithm. A 24×24 estimation window is centred on each block for feature extraction. Hence, each block is shared with its eight immediate neighbours during feature extraction. The model parameters, X_k and Φ_k, are estimated from training sets of samples from each texture class. Other parameters need to preassign are $e = 0.98$, $\rho^2 = 10$, $\nu^2 = 10$ and $\phi = 0.7$.

Figure 2a shows a composite image consisting of three textures selected from Brodatz's photographic album[13]. The textures are cork(D4), wool(D19) and raffia(D84). Fig. 2b shows the classification result when non-spatial ML classifier is used. Note that the percentage of misclassification is relatively high, especially for those blocks in the vicinity of the boundary between wool and raffia. The reason for this is that, for those boundary blocks, the estimated features are mix types. Hence, their characteristics are difficult to predict. Fig. 2c shows the result when the ML spatial classifier is employed. Significant improvements have been achieved, with most of the spurious regions being removed. Fig 2d shows the estimated boundaries after the application of the ML edge estimator. The detected boundaries are close to the boundaries among the texture regions.

Figure 3 shows another example of segmentation of an image which is constructed of two Brodatz textures, grass(D9) and wool(D19). Fig. 3a shows the constructed image with more realistic boundaries between the textures. Fig. 3b and 3c give the classification results when non-spatial classifier and ML spatial classifier are used, respectively. The estimated boundaries are shown in Fig. 3d. Again, the results show the effectiveness and robustness of the proposed algorithm.

Figure 4 shows, as a final example, the segmentation of an outdoor image. Two regions, sky and trees, are contained in the image. Fig. 4b and 4c give, respectively, the spatial classification and edge estimation results.

6. CONCLUSION

A new algorithm for segmenting an image by its textural properties is presented. The proposed method detects texture boundaries by first classifying pixels with similar features into one of the possible texture classes by using a ML spatial classification technique. The exact texture boundaries are estimated in the second stage by a line by line edge estimator. Finally, the estimated edges are processed by a Kalman filter to obtain smooth boundaries. The segmentation results demonstrate the effectiveness and robustness of the developed algorithm.

REFERENCES

1. R. M. Haralick, "Statistical and structural approaches to texture", Proc. IEEE 67, pp. 786-804, 1979.

2. R. W. Conners, M. M. Trivedi and C. A. Harlow, "Segmentation of a high-resolution urban scene using texture operators", Comput. Vision Graphics Image Process. Vol 25, pp. 273-301, 1984.

3. M. Galloway, " Texture analysis using gray-level run lengths", Comput. Vision Graphics Image Process., Vol 4, pp. 172-199, 1974.

4. F. D'Astous and M. E. Jernigan, " Texture discrimination based on detailed measures of the power spectrum", Proc. IEEE Comput. Societ. Conf. on Pattern Recognition and Image Processing, pp. 83-86, 1984.

5. R. L. Kashyap and A. Khotanzard, " A model based method for rotation invariant texture classification", IEEE Trans. Pattern Anal. Machine Intell., Vol. PAMI-8, no. 4, pp. 472-481, July 1986.

6. R. Chellappa and S. Chatterjee, "Classification of textures using gaussian Markov random fields", IEEE Trans. Acoustics, Speech, Signal Process., Vol. ASSP-33 No. 4 pp. 959-963, 1985.

7. Therrien, C. W.,"An estimation-theoretic approach to terrain image segmentation", Comput. Graphics Image Process., Vol. 22, 1983, pp 313-326.

8. Derin, H. and Elliott, H.,"Modeling and segmentation of nosiy and textured images using Gibbs random fields", IEEE Trans. Pattern Anal. Machine Intell., PAMI-9 Jan. 1987, pp 39-55.

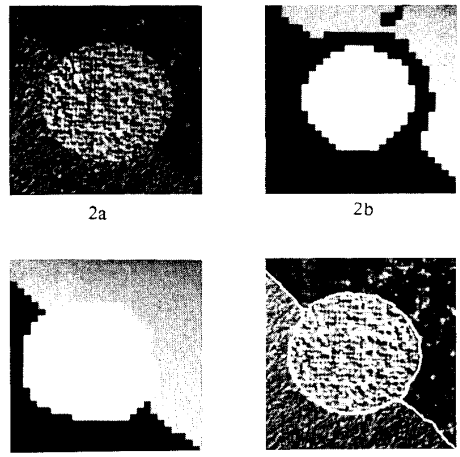
9. Cohen, F. S. and Cooper, D. B.,"Simple parallel hierarchical Markovian random fields", IEEE Trans. Pattern Anal. Machine Intell., PAMI-9, 1987 pp 195-219.

10. Besag, J. E.,"Spatial interaction and statistical analysis of lattice systems", J. Royal Stat. Soc., Ser. B, Vol. 36 1974, pp 192-236.

11. Kashyap, R. L. and Rao, A. R.,"Dynamic stochastic models from empirical data", New York: Academic, 1976.

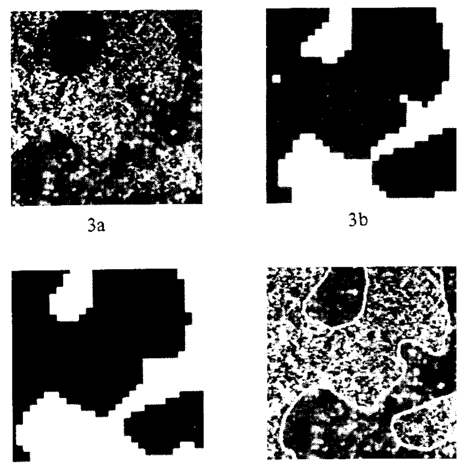
12. Anderson, B. D. and Moore, J. B.," Optimal filtering", Pentice-Hall,. 1979.

13. Brodatz, P.,"Textures: a photographic album for artist and designers", New York: Bover, 1966.



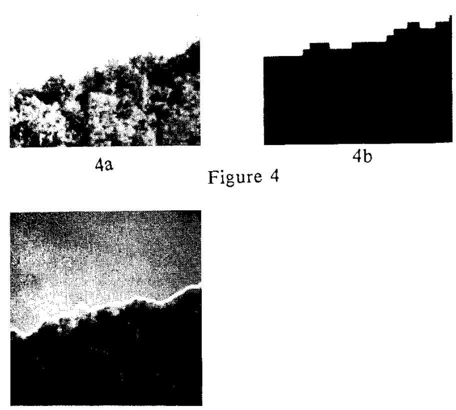
2a 2b 2c

Figure 2



3a 3b 3c 3d

Figure 3



4a 4b 4c

Figure 4